



**INSTITUTO POLITÉCNICO NACIONAL**

**UNIDAD PROFESIONAL INTERDISCIPLINARIA DE  
INGENIERÍA Y CIENCIAS SOCIALES Y ADMINISTRATIVAS**

**LICENCIATURA EN INGENIERÍA EN INFORMÁTICA**

PRESENTAN

- Gonzalez Calzada Maximiliano
- Hernandez Garcia Juan Jose
- Maldonado Martínez Kevin Noel
- Martínez Lagunas Andrik Jeovany
- Muñoz Castro Angel Daniel

DOCENTE

**Ortega Avalos Julia Alicia**

ASIGNATURA

**Seguridad Informática**

SECUENCIA

**4NM42**

12 de noviembre de 2024

---

## 1. Nombre del proyecto

"Web API Security Scanner"

## 2. Objetivo

Desarrollar una herramienta que escanee las API web en busca de vulnerabilidades de seguridad, asegurando que sean sólidas contra amenazas comunes y proporcionando recomendaciones de mejora.

### 2.1. Características Clave

- **Detección Automatizada de Vulnerabilidad:** El escáner debe probar automáticamente las API en busca de vulnerabilidades comunes, incluyendo:
  - **Defectos de autenticación:** Compruebe si hay mecanismos de autenticación débiles o faltantes (por ejemplo, JWT, OAuth).
  - **Problemas de autorización:** Prueba de control de acceso incorrecto (por ejemplo, usuarios que acceden a los recursos que deberían).
  - **Validación de entrada:** Identifique puntos finales vulnerables a la inyección SQL, secuencias de comandos en sitios cruzados (XSS) e inyección de comandos.
- **Limitación de tarifas:** Asegúrese de que las API tengan una limitación de velocidad adecuada para prevenir el abuso (Ataques DDoS).
- **Exposición a Datos Sensibles:** Compruebe si la información confidencial (como contraseñas, números de tarjetas de crédito) se transmite sin cifrado.
- **Informes y Recomendaciones:** Después del escaneo, genere un informe detallado que resuma:
  - Vulnerabilidades detectadas
  - Niveles de riesgo (alto, medio, bajo)
  - Pasos de remediación sugeridos o mejores prácticas.
- **Interfaz amigable para el usuario:** Cree una interfaz de usuario simple que permita a los usuarios ingresar puntos finales y configuraciones de API fácilmente.

### 2.2. Pasos de Implementación

1. **Research Common API Vulnerabilities:** Familiarícese con OWASP API Security Top Ten y otras pautas de seguridad relevantes.
2. **Elija un Lenguaje/Marco de Programación:** Seleccione un idioma (como Python o JavaScript) y un marco adecuado para construir el escáner.

- 
3. **Desarrolle Core Scanning Logic:** Implemente módulos para cada tipo de vulnerabilidad, aprovechando bibliotecas o herramientas cuando corresponda (por ejemplo, OWASP ZAP para integración).
  4. **Construir la interfaz de usuario:** Cree una interfaz web o de línea de comandos que permita a los usuarios ingresar sus detalles de API e iniciar escaneos.
  5. **Pruebas:** Pruebe el escáner contra API vulnerables conocidas para asegurarse de que detecta con precisión las vulnerabilidades.
  6. **Documentación:** Escriba documentación clara para los usuarios sobre cómo usar la herramienta e interpretar los resultados.

### 3. Alcance

#### 3.1. Cobertura de Seguridad:

- **Autenticación y Autorización:** Revisar las configuraciones de autenticación (JWT, OAuth) y control de acceso.
- **Validación de Entrada:** Identificar vulnerabilidades de inyección (SQL, XSS, comandos).
- **Limitación de Tarifas y Exposición de Datos Sensibles:** Comprobar la limitación de tasa para evitar abusos y verificar si se transmiten datos sensibles sin cifrado.
- **Detección de Amenazas Comunes:** Enfocarse en las vulnerabilidades descritas en el OWASP API Security Top Ten.

#### 3.2. Límites del Proyecto:

- **Exclusiones:** La herramienta no se enfocará en realizar pruebas de rendimiento o funcionalidad de la API; su único enfoque será la seguridad.
- **Limitaciones Técnicas:** El alcance inicial del proyecto no incluirá integración avanzada con sistemas de terceros para generación de reportes o monitoreo, aunque esta funcionalidad podría considerarse en versiones futuras.

### 4. Justificación

En la actualidad, las API son fundamentales para el funcionamiento de aplicaciones modernas, permitiendo la comunicación entre diferentes sistemas y servicios. Con el aumento de su uso, también ha crecido la superficie de ataque, haciendo que la seguridad de las API sea una prioridad. A continuación se detallan las razones que apoyan la necesidad de un escáner de seguridad para APIs web:

- 
- **Creciente Dependencia de APIs:** Con la propagación de arquitecturas basadas en microservicios y el uso extensivo de servicios web, las APIs se han convertido en un punto de entrada crucial para las aplicaciones. Esto las convierte en un objetivo atractivo para los atacantes.
  - **Vulnerabilidades Comunes:** Existen vulnerabilidades comunes en las APIs, como las descritas en el OWASP API Security Top Ten. Un escáner automatizado puede ayudar a identificar y mitigar estas vulnerabilidades antes de que sean explotadas.
  - **Automatización y Eficiencia:** Realizar análisis de seguridad manualmente es intensivo en tiempo y recursos. Una herramienta automatizada permite escanear múltiples APIs de forma eficiente, liberando a los equipos de seguridad para que se concentren en análisis más profundos y en la remediación de problemas detectados.
  - **Mejora Continua en la Seguridad:** La generación de informes detallados con recomendaciones proporciona a los desarrolladores y equipos de seguridad una guía para mejorar continuamente la postura de seguridad de sus APIs.
  - **Educación y Conciencia:** Incluir una interfaz amigable y documentación clara no solo hace que la herramienta sea accesible, sino que también educa a los usuarios sobre las prácticas de seguridad y la importancia de proteger sus APIs.
  - **Cumplimiento Normativo:** Muchas organizaciones están sujetas a regulaciones que exigen la implementación de prácticas de seguridad adecuadas. Un escáner de seguridad puede ayudar a demostrar el cumplimiento de estas normativas.
  - **Proactividad ante Amenazas:** La herramienta no solo identificará vulnerabilidades ya existentes, sino que también puede servir como un medio proactivo para detectar configuraciones inseguras y prácticas deficientes en la implementación de APIs.

## 5. Marco teórico

### 5.1. Introducción a las APIs Web

**Definición:** Las APIs (Interfaz de Programación de Aplicaciones) permiten que diferentes aplicaciones se comuniquen entre sí. Son cruciales en la arquitectura de software moderna, especialmente en aplicaciones web y móviles.

**Importancia:** Facilitan la integración y la interoperabilidad, pero también presentan riesgos de seguridad que deben ser gestionados adecuadamente.

### 5.2. Amenazas Comunes a las APIs

- **Inyección SQL:** Permite a los atacantes ejecutar consultas maliciosas en la base de datos.

- 
- **XSS:** Aprovecha la confianza del navegador en las API para ejecutar código en el contexto de un usuario.
  - **Falsificación de solicitudes entre sitios (CSRF):** Permite a un atacante hacer solicitudes en nombre de un usuario autenticado.
  - **Desbordamiento de datos:** Implica enviar un volumen alto de datos a la API con el objetivo de desestabilizar o hacerla inaccesible.

## 6. Planeación

## 7. Planteamiento de la solución